BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 11/25 pp718–729
DOI: 10.26599/BDMA.2023.9020035
Volume 7, Number 3, September 2024

An Evaluation of Variational Autoencoder in Credit Card Anomaly Detection

Faleh Alshameri* and Ran Xia

Abstract: Anomaly detection is one of the many challenging areas in cybersecurity. The anomaly can occur in many forms, such as fraudulent credit card transactions, network intrusions, and anomalous imageries or documents. One of the most common challenges in anomaly detection is the obscurity of the normal state and the lack of anomalous samples. Traditionally, this problem is tackled by using resampling techniques or choosing models that approximate the distribution of the normal states. Variational AutoEncoder (VAE) has been studied in anomaly detections despite being more suitable in generative tasks. This study aims to explore the usage of VAE in credit card anomaly detection and evaluate latent space sampling techniques. In this study, we evaluate the usage of the convolutional network-based VAE model on a credit card transaction dataset. We train two VAE models, one with a large number of normal data and one with a small number of anomalous data. We compare the performance of both VAE models and evaluate the latent space of both VAE models by rescaling them with reconstruction error vectors. We also compare the effectiveness of the VAE model with other anomaly detection models when they are trained on imbalanced dataset.

Key words: anomaly detection; optimization; imbalanced dataset; generative modeling; Convolutional Neural Network (CNN); Variational AutoEncoder (VAE); latent space scaling; reconstruction error

1 Introduction

Cyberattacks on networks and computer systems have risen quickly along with the Internet's explosive growth[1]. The demand for secured online transactions has risen among private industries and government agencies[2]. However, fraud detection remains challenging for most e-commerce organizations. Anomaly detection is a machine learning topic that has received a lot of attention since it is essential to many applications, including network analysis, intrusion detection, fraud detection, malware detection, health monitoring, brain scans, outlier detection in videos, and Internet of Things (IoTs)[3–5].

The significance of anomaly detection stems from the fact that across a wide range of application fields, anomalies in data can yield substantial and even crucial information[3].

This study uses an algorithmic approach that focuses on modeling cardholders' spending behavior and regarding anomalous data as a statistical outlier comparing the normal spending inliers[6].

Variational AutoEncoder (VAE), is a deep learning architecture, it is a special type of autoencoder that were first introduced by Kingma and Welling[7] and Islam et al[8]. VAE belongs to the family of Probabilistic Graphical Models (PGMs). VAE is defined as a directed probabilistic graphical model, which is obtained by approximation of an artificial

• Faleh Alshameri is with School of Business, University of Maryland Global Campus, Adelphi, MD 20783, USA. E-mail: faleh.alshameri@faculty.umgc.edu.
• Ran Xia is with School of Technology and Innovation, College of Business, Innovation, Leadership and Technology, Marymount University, Arlington, VA 22207, USA. E-mail: r0x28181@marymount.edu.
* To whom correspondence should be addressed.
Manuscript received: 2023-07-16; revised: 2023-11-15; accepted: 2023-11-22

neural network to its posterior[1].

Although VAE was invented as a generative model for input reconstruction, recent works have shown the strong potential of VAE in anomaly detection with various alterations added to the original architecture. In data generation, VAE consists of an encoder-decoder structure, where the encoder transforms the input data into a low-dimension latent representation, and the decoder transforms this latent representation back to the same dimension as the input data[7]. The two models complement one another. In order for the generative model to update its parameters during an iteration of expectation maximization learning, the encoder or recognition model provides an approximation to its posterior over latent random variables. The recognition model uses the decoder or generative model as a form of scaffolding to learn accurate representations of the data, including perhaps class labels. According to the Bayes rule, the recognition model roughly corresponds to the generative model's inverse[7].

The intuition of using VAE for anomaly detection is from how it approximates the true distribution from which the observed data is sampled. This assumes that normal and anomalous data are generated from different distributions in anomaly detection. Once a VAE model is adequately trained on the normal data, it learns its distribution and creates a continuous latent space of low-dimensional representation. So naturally, if the anomalous data is passed to this VAE model as input, the reconstruction error would be higher than passing normal data as input. Additionally, the anomalous input in the latent space should form some cluster shapes that differ from the normal input.

The purpose of this research paper is to evaluate the usage of a convolutional VAE on a credit card fraud detection dataset. The evaluation includes training two VAE models separately on the normal and anomalous data. Then compare the performances of both models using confusion matrices and F1-scores. We also evaluate the latent space by examine how anomalous data are represented and clustered in this latent space and the effect of latent space scaling on model performance. A model's ability to divide a dataset into its target is indicated by the latent space for a VAE. It is feasible to generate new samples of data by correctly reading the latent space border[8].

The rest of the paper is organized as the following: Section 2: the literature review section, exploring related works on VAE and anomaly detection. Section 3: the research methodology section, explaining the data source, exploratory data analysis, and model construction. Section 4: the data analysis and results section, evaluates the model performances as well as latent space scaling. And finally, Section 5: the discussion and conclusion section, concludes the paper with findings and further discussions.

2 Literature Review

Observations that differ so greatly from other observations as to raise suspicions that they were produced by distinct mechanisms are known as anomalies, sometimes known as outlier[3, 9, 10]. For many years, anomaly detection has been researched. They can be divided into three categories: supervised, semi-supervised, and unsupervised anomaly detection depending on whether the labels are employed in the training process[3, 4].

Finding prospective time series data patterns that drastically deviate from expected behavior is known as anomaly detection. In their study[11], a temporal convolutional network framework is utilized as a predictor model, and multivariate Gaussian distribution is employed to identify anomaly points in time series. This method of solving the time series anomaly detection problem is known as unsupervised learning.

Liu et al.[4] proposed an online anomaly detection framework for time series, based on temporal convolution networks and Gaussian mixture model. This framework enables the mapping of high-dimensional time series into a low-dimensional feature space, facilitating efficient anomaly detection. The experiment results suggest that the optimized temporal convolutional network is capable of extracting salient and discriminative features from time series data. Additionally, the Gaussian Mixture Model (GMM) demonstrates strong generalization and reliability in detecting anomalies. GMM with Bayesian inference enables effective anomaly detection, maintaining a low False Positive Rate (FPR) 0.796%, without compromising on high recall rate 99.67%. Its applicability extends to time series anomaly detection.

Lorgat et al.[12] proposed a general framework for network traffic data anomaly detection, as a type of time-series data with four distinctive characteristics: self-similarity, long-range dependence, non-Gaussian,

and non-stationarity.

He and Zhao[11] experimented with a Temporal Convolutional Network (TCN) for anomaly detection. They showed that temporal convolutional networks could automatically learn inherent patterns in sequential data. The TCN are tested on different real-world datasets, electrocardiograms (ECG), space shuttle and 2D gesture. TCN-based approach works well on three datasets. The results show higher precision and F1-score, 0.930 and 0.901, respectively, for the ECG dataset.

Comparing disparities between generated data and origin data is the fundamental detection premise of VAE-based approaches. In the case of test data, a high likelihood exists that the test data and training data will most likely belong to the same class if the difference is minor, indicating that the test data is normal. The test data may belong to the opposite class of training data if the difference is significant; in this case, the test data is probably anomaly data[13].

Utilizing the reconstruction probability obtained from the VAE, An and Cho[9] presented a method for detecting anomalies. By taking into consideration the idea of variability, the reconstruction probability combines the VAE's probabilistic traits. The suggested method outperforms autoencoder-based and principal-component-based algorithms, according to experimental results. By utilizing the variational autoencoder's generative properties, it is possible to rebuild the data and understand the anomaly's underlying causes.

Zhang et al.[13] proposed a new feature encoding technique based on Probability Mass Function (PMF). This encoding technique helps generative models with handling categorical features. They also proposed a new detection method based on Adversarially Learned Inference (ALI). The experiment results demonstrate that their method possesses the benefits of superior detection accuracy and efficient training, and exhibits the capability to identify unfamiliar attacks. Table 1 shows the comparison of their methods with Support Vector Machine (SVM) based on binary classification, standard One-Class SVM (OC-SVM), One-Class SVM with 3 features selected in advance (OC-SVM-3), and Generative Adversarial Network (GAN) based method.

The evaluation indicators are the detection accuracy F1-measure, the higher value indicates the higher accuracy. The second evaluation indicator is the train time and test time, the shorter time indicates the higher efficiency. Table 1 shows that their method (ALI) outperforms GAN in both F1-measure and efficiency. OC-SVM-3 shows the best performance and efficiency, but this approach involves manual feature pre-selection.

Table 1 Experiment results of five methods.

Method	F1-measure	Train time (s)	Test time (s)
SVM	0.8318	37.9	1.4
OC-SVM	0.0078	940.5	257.5
OC-SVM-3	0.9691	9.9	1.8
GAN	0.9247	1223.0	7291.0
ALI	0.9602	980.3	6.8

Islam et al.[8] attempted to overcome the drawbacks of conventional synthetic oversampling techniques, by proposing to augment crash data using VAE. The ability to accurately choose the decision boundary during data production with VAE can aid in lowering the overfitting that is present in synthetic oversampling techniques.

Ahn et al.[14] evaluated various deep learning based algorithms for finding anomalies in a spaceship attitude control system. The study seeks to demonstrate the viability of the suggested anomaly detection method based on the chosen neural network model for assessing the spacecraft attitude control system's condition.

Kingma and Welling[7] proposed a novel deep neural network, VAE, used for sequential data encoding and decoding. Such a model improves the discreteness of a simple GMM approach in encoding an input vector.

Pangione et al.[15] suggested a technique for detecting anomalous data from a robot while training with a small amount of nominal data. They employed VAE in their study to find abnormalities in the robotic glovebox configuration. The extremely intricate and structured nature of the relationship between the measured signals and the health of the robot serves as the driving force behind this decision.

Guo et al.[3] published a GRU-based Gaussian mixture VAE for unsupervised anomaly detection. The paper shows that simply adding temporal data processing ability to the architecture can make the VAE model recognize long-term dependencies from data.

Ahn et al.[14] used deep generative models to detect anomalies and characterize spacecraft attitude control

systems failures. They found that VAE can generate abnormal samples rather than normal samples even the model is only trained on normal data. They attributed this observation to the unclearness of normal and abnormal data due to the noise caused by the reconstruction process from the sampling of the reparameterization layer of the VAE model.

There has been many studies focus on demonstrating VAE anomaly detection usages or adjustment with temporal features since Kingma and Welling original[7]. Unlike TCN or GRU-based Gaussian mixture AVE approaches, our study discerns anomaly detection from non-temporal perspective, as not all anomaly detection problems can be framed as time-series. Furthermore, we recognize the noise caused in the VAE reconstruction process may clutter the separation of normal and anomaly data. So, we propose a way of rescaling VAE latent space and evaluate its effect on performance.

3 Research Methodology

3.1 Data description

The dataset used in this study is retrieved from kaggle.com, an open-source project and dataset sharing website. This dataset is collected and analyzed during a research collaboration of worldline and the machine learning group (http://mlg.ulb.ac.be) of ULB (Université Libre de Bruxelles, Belgium) on big data mining and fraud detection. The dataset contains transactions made by credit cards in September 2013 by European cardholders. This dataset contains 284 807 transactions and thirty numeric features. This dataset is highly imbalanced. The positive class (492 frauds transactions) is 0.172% of all transactions. Due to confidentiality issues, the dataset has been processed with Principal Component Analysis (PCA) transformation, and features are named from V1 to V28 as the principal components obtained from PCA transformation. In addition to these principal components, "time" and "amount" features are not transformed.

3.2 Research framework overview

Figure 1 shows the framework overview of the VAE training stage. In this stage, we first filter the features from the dataset to only focus on the most discriminant features, then create the normal and anomaly training set that are randomly sampled from the original dataset. The normal versus anomaly sample size is highly imbalanced. Then we train two VAE models. The first model (VAE-normal) is trained only with the normal samples, and other one (VAE-anomaly) is trained only with anomaly samples. More details of feature selection are shown in Section 3.3.

Figure 2 shows the evaluation stage of the framework that tests both VAE models on the same dataset with balanced class ratio. In the evaluation, we first evaluate the classification performance of both VAE models by thresholding the reconstruction error that is optimized on F1-scores. We then rescale the latent space by the reconstruction error to observe the effect on classification performance. Section 3.4 provides more explanation of data sampling and model construction.

In Figs. 1 and 2, the latent space is denoted as L, the Rescaled latent space is denoted as L_{new}. We denote the model reconstruction errors as E, for the VAE-normal

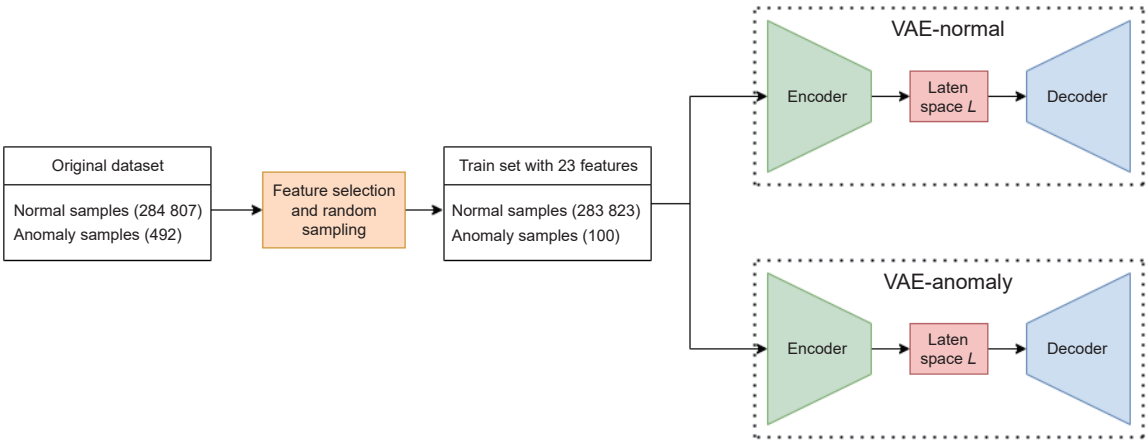


Fig. 1 VAE model training overview.

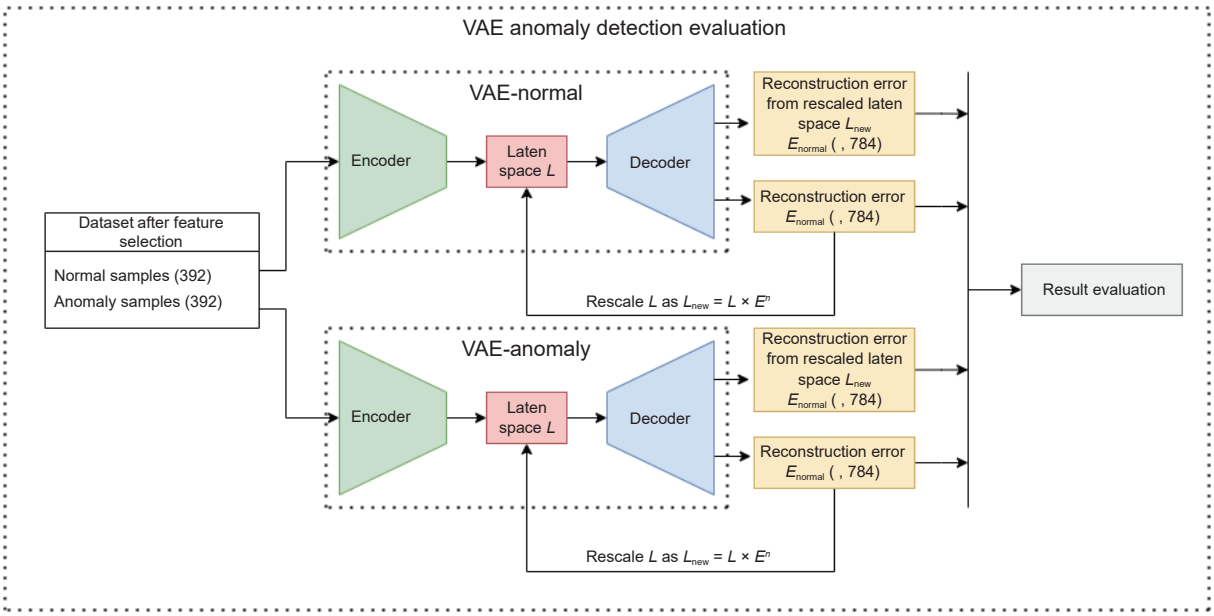


Fig. 2 VAE model evaluation and latent space rescaling overview.

and VAE-anomaly models, these reconstruction error are denoted as E_{normal} and E_{anomaly}, respectively. We also denote the scaling coefficient as n. So the latent rescaling is denoted as $L_{\text{new}} = L \times E^n$, where n is the rescaling coefficient.

3.3 Data exploration and feature selection

Before training the model, it is necessary to confirm sufficient discriminations between the normal and the anomalous data. Two key factors need to be examined, the feature collinearity test and the Two-Sided Kolmogorov-Smirnov (KS) test. The collinearity test explores relationships amongst features for both classes. This could help determine if a feature pair behaves differently in the anomalous data and hence provide rationale in feature engineering. The Two-Sided Kolmogorov-Smirnov test is a statistical test to determine if two underlying one-dimensional probability distributions are different. Every feature from the normal dataset is tested against the anomalous dataset in this study.

We plot two correlation heatmaps of both normal and anomalous classes to examine the collinearity. Figure 3 shows the heatmaps between the two classes.

Figure 3 shows that the normal class has consistently low negative correlations among the features, while the anomalous class exerts significantly higher positive correlations with more significant variation. It also shows the V1 through V28 and amount features. Dark blue (1.00) indicates strong positive correlations, whereas dark yellow indicates strong negative correlations (−1.00). V1 to V18 show the most significant variation within the anomalous class. At this step, it can be confirmed that the two classes have distinctive feature collinearity and some discriminative characteristics.

The second test is the Two-Sided KS test to compare features between normal and anomalous classes. Figure 4 shows the results.

Because a VAE model tries to approximate the distribution from which samples are drawn, it is also vital to examine if the feature distributions are different in the anomalous class. This can be done by visualizing the distribution plots and the Two-Sided Kolmogorov-Smirnov test. As the normal data are significantly larger than the anomalous data, the Two-Sided Kolmogorov-Smirnov test is performed on 492 normal samples randomly from the normal data and all 492 anomalous data. The visualizations in Fig. 4 show the distribution of each feature in both normal and anomalous classes. Most of the features have nearly zero Kolmogorov-Smirnov value, which fails to reject the null hypothesis that the two distributions, normal and anomalous, are the same. However, features V13, V15, V22, V24, V25, and V26 show significantly larger Kolmogorov-Smirnov values, and their distributions are very close. These features are highlighted in red in Fig. 4.

Based on this exploratory analysis, it is apparent that the anomalous class significantly differs from the

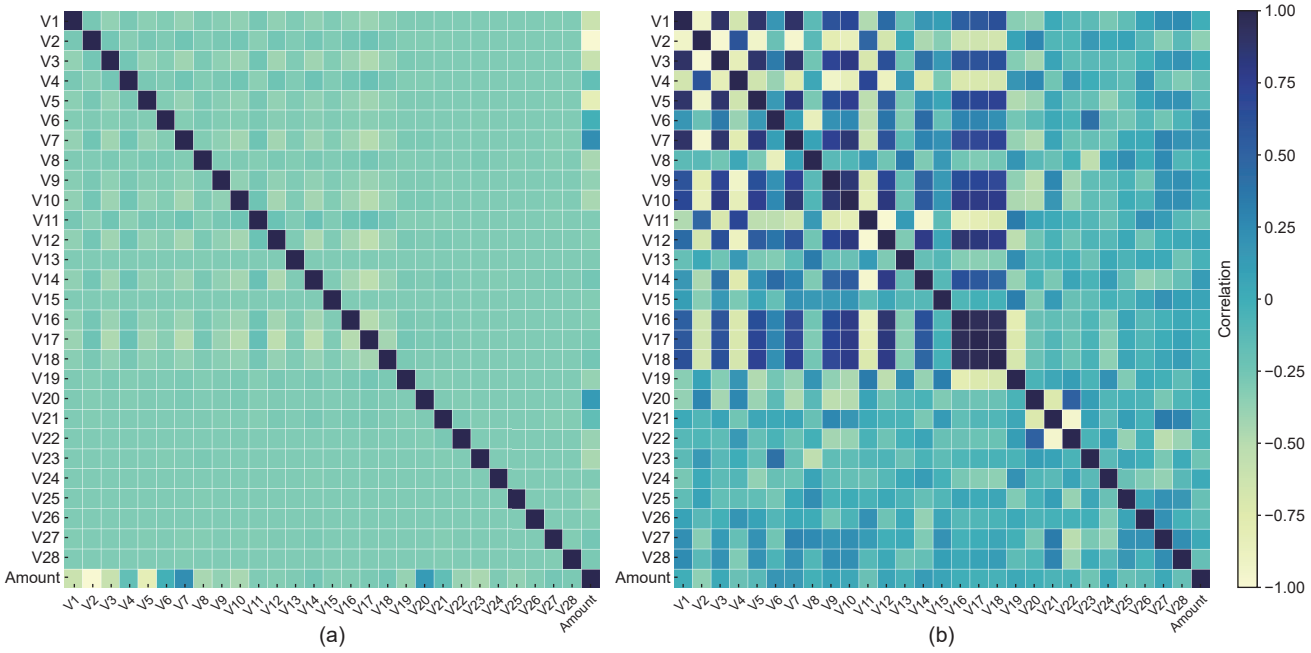


Fig. 3 Heatmaps of (a) normal class and (b) anomalous class.

normal class. Additionally, because of the similar distributions of V13, V15, V22, V24, V25, and V26, these features are removed from the training and testing dataset as they are noise to the VAE model. Total of 23 features are selected, including V1 to V12, V14, V16 to V21, V23, V27, V28, and amount.

3.4 Model construction and experiment methods

As with other generative models, the VAE uses latent variables and models the input data distribution by approximating the joint probability distribution[7]. This approximation is made by using neural networks. Convolutional Neural Network (CNN) is one of the popular deep learning algorithms primarily seen in its usage in computer vision and image processing. As the name stated, a CNN-based model performs discrete convolution operations at each layer using a square filter matrix. In computer vision and image processing, CNN has been proven to capture the spatial and temporal dependencies in images[11]. In our case, we choose CNN as the base of the VAE model, so that high-level characteristics from both normal and anomalous data can be captured and sent to the next level of the model and eventually reflected in the latent space. Also, using CNN requires fewer data preprocessing, and the total training parameters are reduced due to the convolution operations in each layer. In this study, we evaluate two similar model. One of the VAE models is trained on normal data (VAE-normal), and the second is trained on anomalous data (VAE-anomalous). Figure 5 shows the structures of the encoder and decoder of both VAE models.

We sample 100 anomalous data out of the 492 anomalous transactions. Then, we sample 392 normal data and combine them with the rest of the 392 anomalous data. This balance-ratio dataset (392 normal and 392 anomalous) is preserved for the evaluations of both models. The first VAE model (VAE-normal) is trained on the rest of the normal dataset (283 823 samples), and the second VAE model (VAE-anomaly) is trained on the 100 anomalous data.

While the VAE loss function consists of reconstruction error and KL-Divergence loss, we only used the reconstruction error for the anomaly detection task. The reconstruction error used in the anomy detection part is the Root Mean Square Error (RMSE) between the input and the reconstructed vector, $\text{rmse} = \sqrt{\frac{\sum (v_1 - v_2)^2}{N}}$, where v_1 is the input vector, and v_2 is the reconstructed vector. To detect an anomaly, we pass both training and testing datasets to a trained VAE model and receive two lists of reconstruction errors. For example, if the VAE-normal is only trained on the normal dataset, passing anomalous data to the model causes a higher reconstruction error. We then find a threshold by setting the reconstruction error at a

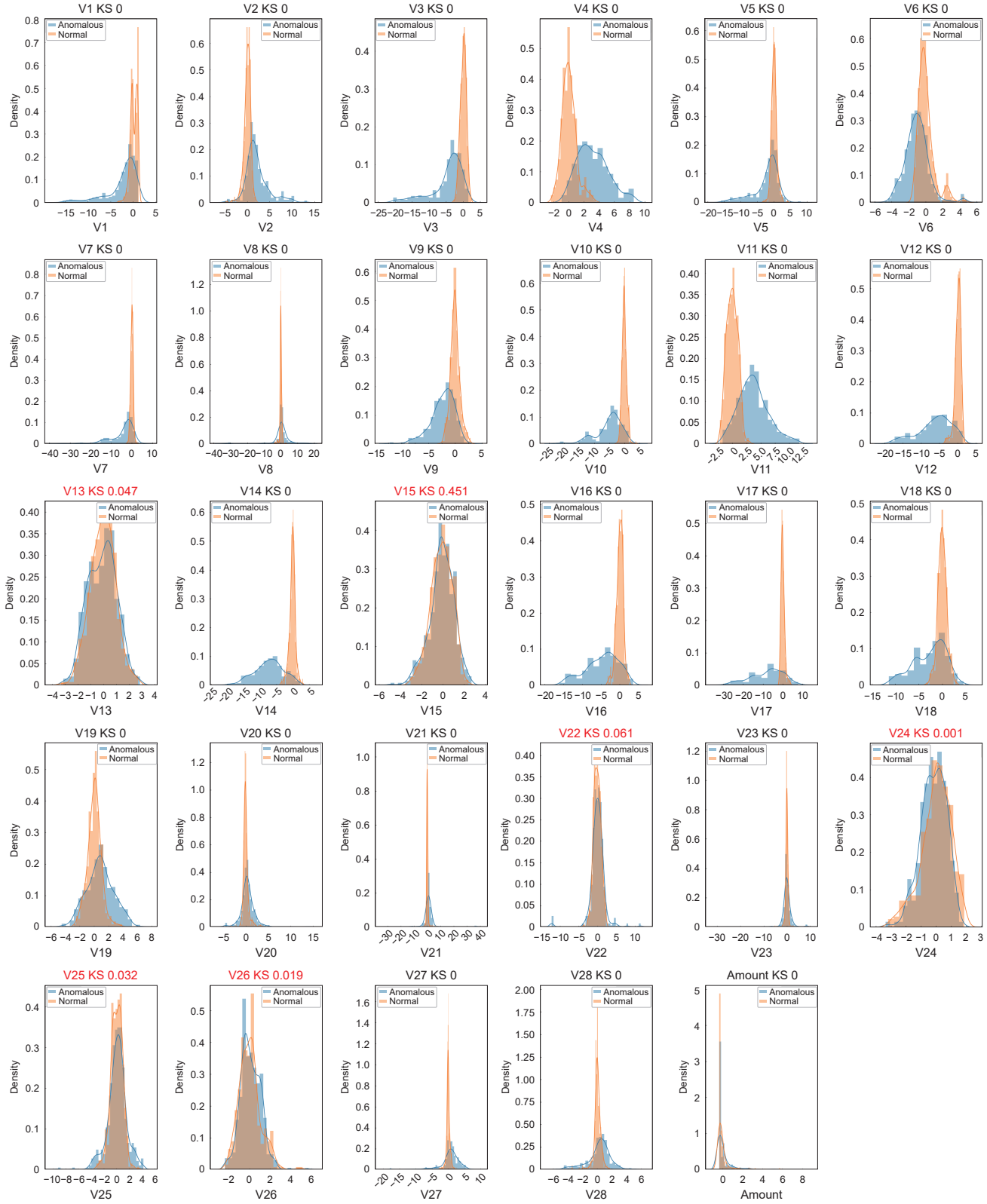
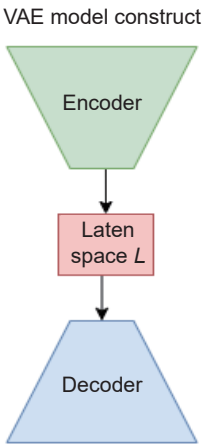


Fig. 4 Visualizations of the feature distributions between the normal class and the anomalous class, features in red show high Kolmogorov-Smirnov value.

selected quartile in the training dataset reconstruction errors list. We then use this threshold in the list of testing dataset reconstruction error and evaluate the F1-score. The threshold that results in the highest



Layer (type)	Output shape	Number of parameters
encoder_input (InputLayer)	[(None, 23, 1)]	0
encoder_conv_0 (Conw1D)	(None, 23, 16)	96
encoder_conv_1 (Conv1D)	(None, 12, 32)	2592
encoder_conv_2 (Conv1D)	(None, 12, 64)	10 304
flatten_41 (Flatten)	(None, 768)	8
dense_85 (Dense)	(None, 32)	24 608
latent_mu (Dense)	(None, 3)	99
latent_log_var (Dense)	(None, 3)	99
sampling_41 (Sampling)	[(None, 3), (None, 3)]	0
decoder_input (InputLayer)	[(None, 3)]	0
dense_87 (Dense)	(None, 768)	3072
reshape_45 (Reshape)	(None, 12, 64)	0
decoder_conv_0 (Conv1DTRansp)	(None, 12, 64)	20 544
decoder_conv_1 (Conv1DTRansp)	(None, 24, 32)	10 272
decoder_conv_2 (Conv1DTRansp)	(None, 23, 1)	161
cropping1d_41 (Cropping1D)	(None, 23, 1)	0

Fig. 5 Structures of the encoder and decoder of the CNN based VAE model.

F1-score is then selected. F1-score is calculated as

$$\text{F1-score} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})},$$

where TP is the true positive rate, FP is the false positive rate, and FN is the false negative rate.

We also examine how rescaling the latent space affects the model performance. We rescale the latent distribution by the reconstruction error: $L_{\text{new}} = L \times E^n$. We use L_{new} in the same anomaly detection process mentioned above for both VAE-normal and VAE-anomaly. We then compare the results from using L_{new} to the results using the original L.

4 Data Analysis and Result

4.1 Model performance and latent space visualization

We evaluate the models by both model performance and the latent space robustness. To evaluate the model performance, we use the confusion matrix and the F1-score. To evaluate the latent space, we visualize the latent distribution of the encoded training input and the testing input. Figure 6 shows the results. In the classification report, the "normal" class is the non-fraud transactions in dataset, and it is negative in model prediction. Reversely, "anomaly" class is the fraud transactions in the dataset, and it is positive in model prediction. In the confusion matrix, true negative count is the number of ground-truth non-fraud samples being predicted as non-fraud by model. Similarly, true positive count is the number of ground-truth fraud samples being predicted as fraud. False negative count is the number of ground-truth fraud samples being predicted as non-fraud by model, and false positive count is the number of ground-truth non-fraud samples being predicted as fraud.

Our model achieved an overall 0.92 F1-score with a slightly high false negative rate. As the latent space of VAE is continuous, the latent representation of the normal and anomalous samples can not be linearly or near linearly separated. So, when those in the overlapping area of the latent space are likely to be misclassified. In our anomaly detection case, specific anomalous samples are very similar to the normal samples. The following 3D latent space visualization further illustrates it.

In Fig. 7, the green group is the normal testing samples in the latent space, and the red group is the anomalous testing samples. Notice that the green group

Class	Precision	Recall	F1-score	Total samples
Normal (non-fraud)	0.89	0.96	0.92	392
Anomaly (fraud)	0.96	0.88	0.91	392
Accuracy	–	–	0.92	784
Macro average	0.92	0.92	0.92	784
Weighted average	0.92	0.92	0.92	784

Class	Precision	Recall	F1-score	Total samples
Normal (non-fraud)	0.89	0.96	0.92	392
Anomaly (fraud)	0.96	0.88	0.92	392
Accuracy	–	–	0.92	784
Macro average	0.93	0.92	0.92	784
Weighted average	0.93	0.92	0.92	784

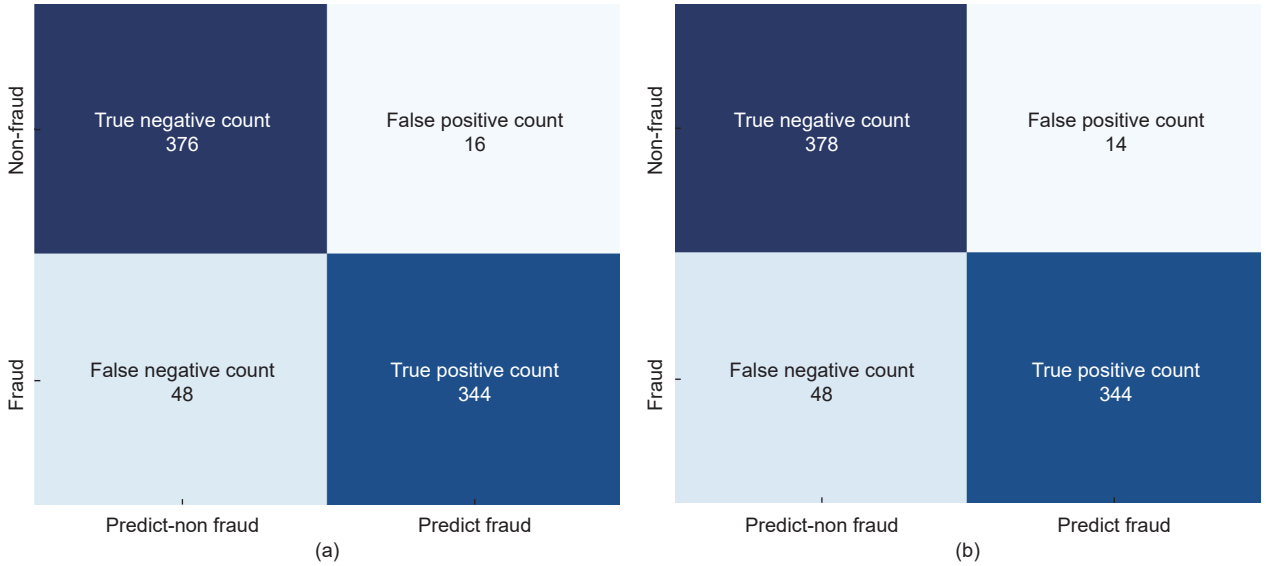


Fig. 6 Classification report and confusion matrix of VAE-normal before (a) and after (b) latent space rescaling.

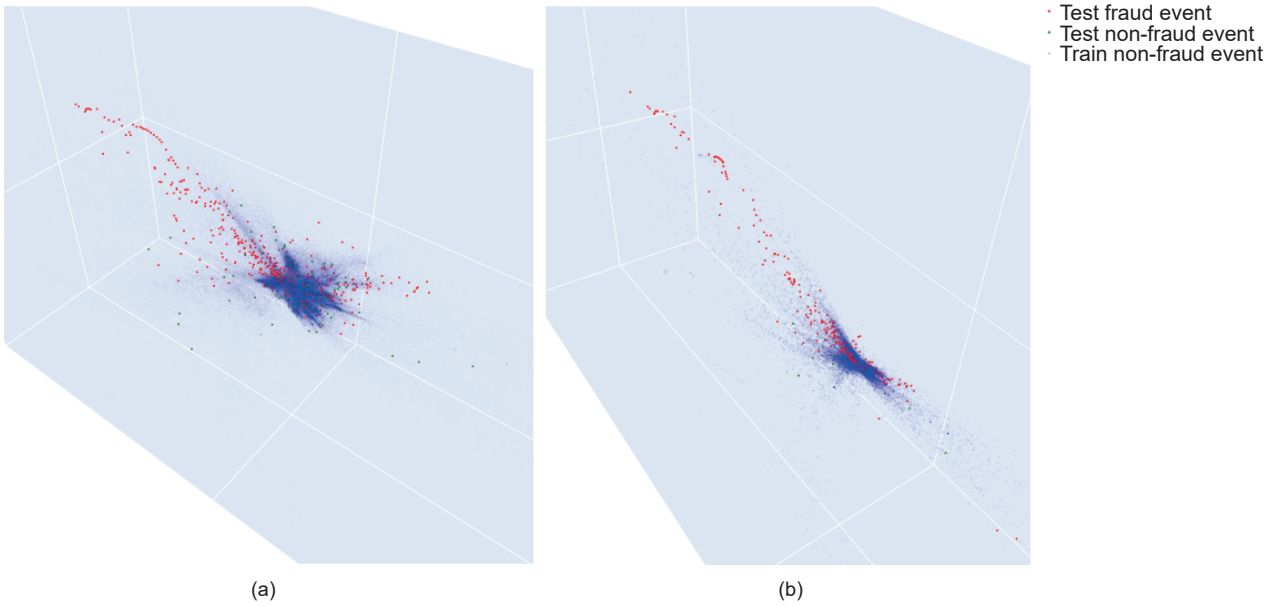


Fig. 7 VAE-normal latent space before (a) and after (b) rescaling.

overlaps with the normal training samples in the "center" of the latent space. This means our CNN-based VAE model can approximate the training data distribution as the latent representation for both normal training and testing from overlapping clusters. On the other hand, the red group also forms a cluster in the latent space, and this means our model also captures the similar characteristics of the anomalous class.

The other evaluation involves testing our model's stability by scaling the latent space with the reconstruction error vector. For a well-trained model, this means the center of the latent space becomes denser as the model is trained on normal samples and the reconstruction error for those samples is low. So,

most of those normal samples will be pulled towards the center of the latent space. Conversely, anomalous testing samples in the latent space would become more spread out due to high reconstruction error. This scaling can better separate the latent space and expose samples that are prone to misclassification; however, if the model is not trained correctly or not able to approximate the distribution that generates the training data, scaling the latent space to become more evenly spread around the center and loses the clustering characteristics as before scaling.

Figure 7 shows the results from latent space scaling with $n = 1$. The normal samples from the testing dataset (green) are significantly more clustered around the centroid of the latent space (covered by the blue group). Moreover, the anomalous samples from the testing dataset (red) are more spread out in the latent space without losing their cluster shape. We then use this rescaled latent space for anomaly detection. Figure 7 shows a minimal improvement of false positive rate after rescaling the latent space. As expected, this is due to the better separation of the normal and anomalous data in the latent space after rescaling. We observe similar results with the VAE-anomaly model.

As shown in Fig. 8, the VAE-anomaly model's false negative rate improves, and false positive rate worsens. The overall F1-score increased by 1% after rescaling the latent space with the error vector. Unlike the VAE-normal model, VAE-anomaly is only trained on 100 anomalous samples, and training a complex model on a small dataset tends to be underfitting. However, this CNN-based VAE model still reached high F1-score. Moreover, the model is improved by rescaling the latent space. Figure 9 shows the visualizations of the latent space before and after rescaling.

4.2 Comparison with other methods

As the study focuses on evaluating VAE on extremely imbalanced dataset, we compare our results with other experiments from data resampling effort and final performance metrics. Most experiments on this dataset attempt to first solve the data imbalance issue. Recent attempts include over-sampling[16, 17], under-sampling[17]. and using synthetic minority oversampling technique (namely SMOTE) to oversample with synthetic data[18]. Table 2 shows the comparison results with other methods. We highlight our model as "VAE-normal", "VAE-anomaly", "VAE-normal-rescaled", and "VAE-anomaly-rescaled".

From Table 2, the VAE-anomaly with rescaled latent space achieves the highest precision, and XGBoost

Class	Precision	Recall	F1-score	Total samples
Normal (non-fraud)	0.87	0.97	0.92	392
Anomaly (fraud)	0.97	0.85	0.91	392
Accuracy	–	–	0.91	784
Macro average	0.92	0.91	0.91	784
Weighted average	0.92	0.91	0.91	784

Class	Precision	Recall	F1-score	Total samples
Normal (non-fraud)	0.89	0.96	0.93	392
Anomaly (fraud)	0.96	0.88	0.92	392
Accuracy	–	–	0.92	784
Macro average	0.93	0.92	0.92	784
Weighted average	0.93	0.92	0.92	784

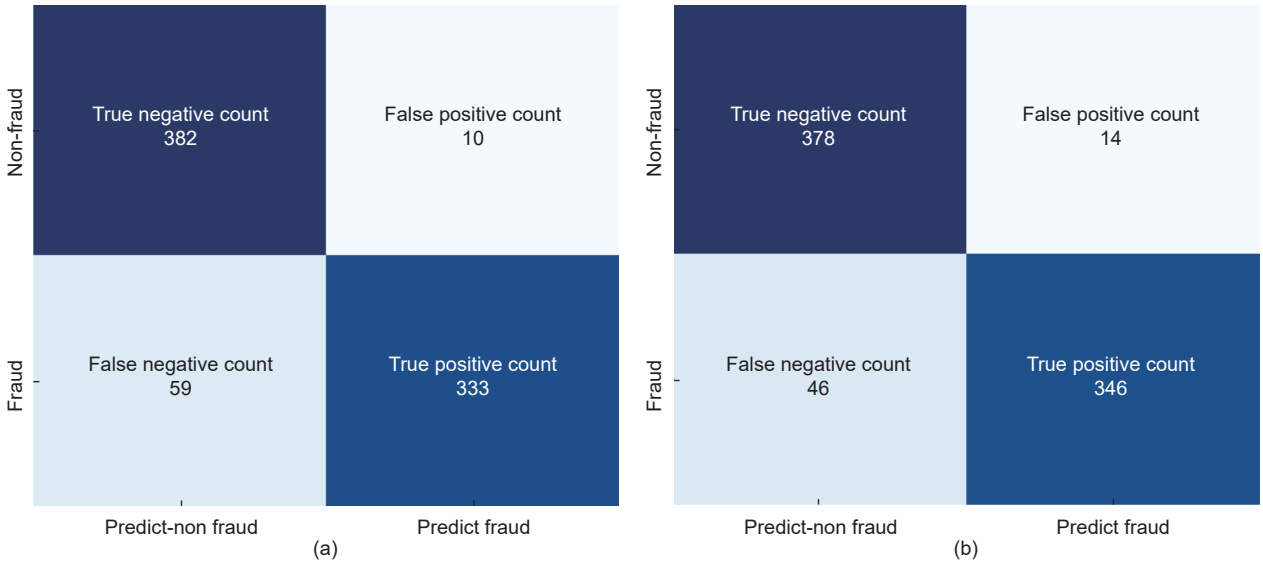


Fig. 8 Classification report and confusion matrix of VAE-anomaly before (a) and after (b) latent space rescaling.

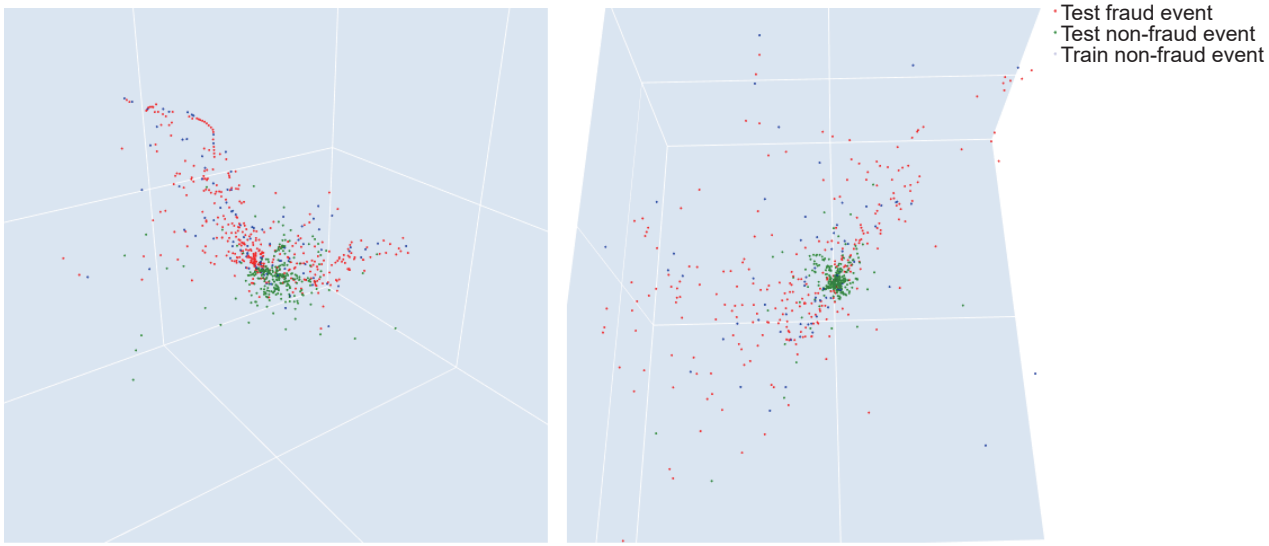


Fig. 9 VAE-anomaly latent space before (a) and after (b) rescaling.

Table 2 Comparison results with other methods.

Model	Resampling method	Precision	Recall	F1-score
VAE-normal	None	0.92	0.92	**0.92**
VAE-anomaly	None	0.92	0.91	0.91
VAE-normal-rescaled	None	0.92	0.92	**0.92**
VAE-anomaly-rescaled	None	**0.93**	0.92	**0.92**
Logistic regression	Under-sample	0.61	0.83	0.70
Complement naïve Bayes	SMOTE	–	–	0.73
KNN	SMOTE	–	–	0.73
SVM	SMOTE	–	–	0.75
Random forest	SMOTE	–	–	0.75
XGBoost	Under-sample	0.53	**0.94**	0.54
XGBoost	Over-sample	0.70	0.93	0.77

with under-sampling achieves the highest recall. Overall, the VAE models from this study achieve the highest F1-scores. Note that the VAE models achieve this performance without requiring any data augmentation or resampling methods. Both VAE-normal and VAE-anomaly achieve similar results suggesting that the model can still detect anomaly even if it is trained only the normal data.

5 Discussion and Conclusion

In this study, we construct two CNN-based VAE models, VAE-normal and VAE-anomaly, and evaluate the latent space of both models. We also experiment with the effect of rescaling the latent space with the reconstruction error vector. The VAE-normal model achieves 0.92 F1-score for both unscaled and scaled latent space. The VAE-anomaly model achieves 0.90 F1-score with unscaled latent space and 0.92 F1-score with scaled latent space. We observe some potential improvements in latent space scaling, especially when training data is insufficient, and this could make the VAE model more suitable for classification tasks.

Using CCN-based VAE models reduces the effort required for data labeling and data preprocessing. It also bypasses the imbalance problem in most anomaly detection scenarios. Additionally, visualization of the latent space provides insights into how the model learns and predicts, thus making the model more explainable and trustworthy.

One limitation of using VAE in anomaly detection is the complexity of analyzing the continuous latent space. The latent space provides a low-dimensional

representation of the input yet is not focused enough on aggregating them by the distinctive characteristics. Our study find that even the majority of the anomalous inputs aggregate in the latent space, they are still not separated enough from the central cluster of the normal input. We find that rescaling the latent space positively affects model performance, especially when training data is insufficient.

For future research, we recommend testing some clustering-based sampling methods in the latent space to better separate the low reconstruction error region and high reconstruction error region in the complex and non-linear latent space.

References

[1] S. Zavrak and M. İskefiyeli, Anomaly-based intrusion detection from network flow features using variational autoencoder, *IEEE Access*, vol. 8, pp. 108346–108358, 2020.

[2] B. Lebichot, F. Braun, O. Caelen, and M. Saerens, A graph-based, semi-supervised, credit card fraud detection system, in *Proc. of the 5th International Workshop Complex Networks & Their Applications*, Milan, Italy, 2016, pp. 721–733,

[3] Y. Guo, W. Liao, Q. Wang, L. Yu, T. Ji, and P. Li, Multidimensional time series anomaly detection: A gru-based gaussian mixture variational autoencoder approach, in *Proceedings of the 10th Asian Confernce on Machine Learning Research*, Beijing, China, 2018, pp. 97–112.

[4] J. Liu, H. Zhu, Y. Liu, H. Wu, Y. Lan, and X. Zhang, Anomaly detection for time series using temporal convolutional networks and gaussian mixture model, *Journal of Physics*: *Conference Series*, vol. 1187, no. 4, pp. 042111–042121, 2019.

[5] G. S. Chadha, I. Islam, A. Schwung, and S. X. Ding, Deep convolutional clustering-based time series anomaly detection, *Sensors*, vol. 21, no. 16, p. 5488, 2021.

[6] M. Soleh, E. R. Djuwitaningrum, M. Ramli, and M. Indriasari, Feature engineering strategies based on a One-point Crossover for fraud detection on Big Data Analytics, *J. Phys.*: *Conf. Ser.*, vol. 1566, no. 1, p. 012049, 2020.

[7] D. P. Kingma and M. Welling, An introduction to variational autoencoders, *Found. Trends® Mach. Learn.*, vol. 12, no. 4, pp. 307–392, 2019.

[8] Z. Islam, M. Abdel-Aty, Q. Cai, and J. Yuan, Crash data augmentation using variational autoencoder, *Accid. Anal. Prev.*, vol. 151, p. 105950, 2021.

[9] J. An and S. Cho, Variational autoencoder based anomaly detection using reconstruction probability, http://dm.snu.ac.kr/static/docs/TR/SNUDM-TR-2015-03.pdf, 2023.

[10] Z. Niu, K. Yu, and X. Wu, LSTM-based VAE-GAN for time-series anomaly detection, *Sensors*, vol. 20, no. 13, p. 3738, 2020.

[11] Y. He and J. Zhao, Temporal convolutional networks for anomaly detection in time series, *Journal of Physics, Conference Series*, vol. 1213, no. 4, pp. 042050–042056, 2019.

[12] M. Lorgat, A. Baghai-Wadji, and A. McDonald, Towards a general framework for network traffic time series anomaly detection, in *Proceedings of the 16th European Conference on Cyber Warfare and Security*, Dublin, Ireland, 2017, pp. 252–260.

[13] L. Zhang, W. Yang, H. Gan, M. Li, X. Wang, and G. Liang, Anomaly detection based on PMF encoding and adversarially learned inference, *Journal of Physics, Conference Series*, vol. 1187, no. 5, pp. 052037–052047, 2019.

[14] H. Ahn, D. Jung, and H.-L. Choi, Deep generative models-based anomaly detection for spacecraft control systems, *Sensors*, vol. 20, no. 7, p. 1991, 2020.

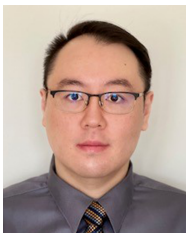
[15] L. Pangione, G. Burroughes, and R. Skilton, Variational AutoEncoder to identify anomalous data in robots, *Robotics*, vol. 10, no. 3, p. 93, 2021.

[16] E. Bilir, UnderSample vs OverSample, https://www.kaggle.com/code/emirbilir/undersample-vs-oversample, 2023.

[17] A. Wiratma, MIE213311-Tugas 1-Logistic & Random Tree-CC Fraud, https://www.kaggle.com/code/arkalilangwiratma/mie213311-tugas-1-logistic-random-tree-cc-fraud, 2023.

[18] F. Alshameri and R. Xia, Credit card fraud detection: An evaluation of SMOTE resampling and machine learning model performance, *Int. J. Bus. Intell. Data Min.*, vol. 23, no. 1, pp. 1–13, 2023.

Faleh Alshameri is a professor of computer information systems. He received the PhD degree from George Mason University, USA. His research interests include text mining, image mining, data science, and big data analytics. He has published several research papers in refereed journals and international conferences.



Ran Xia is a data scientist and engineer. He received the MEng degree in information technology from Marymount University, USA. His research focuses on deep learning and explainable AI in computer vision, natural language processing, and anomaly detection. He has published number of research papers in refereed journals and conferences.